

## Acercando la academia al mundo real: una experiencia de aplicación de Metodologías Agile al proceso de enseñanza-aprendizaje en una asignatura de desarrollo de software

Fabián Tesei<sup>1\*</sup>, Matías Cabrera<sup>1\*</sup>, Marcelo Vaquero<sup>1</sup>, Daniel Tedini<sup>1</sup>  
{FabianElio.Tesei, MatiasAlejandro.Cabrera}@Alumnos.uai.edu.ar,  
{Marcelo.Vaquero, Daniel.Tedini}@uai.edu.ar

<sup>1</sup>Centro de Altos Estudios en Tecnología Informática - Universidad Abierta Interamericana

\*Los autores contribuyeron en igual medida en el presente trabajo.

**Resumen.** Se presenta una experiencia dentro del ámbito académico orientada al desarrollo de un proyecto de software utilizando metodologías ágiles (Agile) bajo la aplicación de Scrum, implementada en la asignatura Trabajo de Campo de la carrera Ingeniería en Sistemas Informáticos de la Facultad de Tecnología Informática de la Universidad Abierta Interamericana. Se describe la propuesta y aplicación de la metodología Agile, así como los roles, herramientas y ceremonias utilizadas. Finalmente, se detalla el resultado de la experiencia evidenciada en indicadores cuantitativos, cualitativos y conclusiones recabadas a partir de los actores involucrados en el proceso.

### 1 Introducción

Dada la vorágine de cambios contemporáneos donde la revolución digital ha acelerado su expansión e impacto, el nuevo modelo de negocio de tecnología informática va de la mano de la velocidad de adaptación [1]. En lo que a desarrollo de software se refiere, una característica de valor importante en esta nueva era de información es la adaptación a los tiempos del mercado [2]. Esta adaptación es crucial para dar a luz un producto, y en este sentido, las metodologías ágiles o Agile<sup>1</sup> constituyen un acercamiento más flexible, más escalable y productivo del desarrollo. Estas metodologías buscan la generación de valor desde el inicio y la entrega rápida de mejoras al negocio en forma iterativa [3]–[5].

Por otra parte, es creciente en los últimos años el interés en la incorporación de las nuevas tecnologías en el proceso de enseñanza-aprendizaje dentro del ámbito académico [6]–[8]. En particular, existe un proceso de búsqueda de

---

<sup>1</sup> Usaremos en este trabajo la denominación en inglés “Agile” por ser la más utilizada en la industria.

la alineación de las asignaturas universitarias con la actualidad del negocio de desarrollo de software y su aplicación [9], [10]. El desarrollo de este trabajo está motivado especialmente a partir de la experiencia de los dos primeros autores, quienes además de su trabajo académico participan en el mercado laboral, trabajando activamente en compañías de desarrollo de software a nivel global<sup>2</sup>. Estas compañías aplican en su mayoría metodologías Agile acompañando las necesidades actuales del negocio [11]. Entendiendo esta realidad y su necesidad creciente en el mercado global, es necesario estudiar la necesidad de su incorporación en el proceso de enseñanza-aprendizaje, en particular en las asignaturas que incluyen el desarrollo de software en sus contenidos, dentro de las carreras que forman profesionales en tecnologías de la información.

Por lo expuesto anteriormente y teniendo en cuenta el rol fundamental de la universidad en la formación de profesionales, presentamos en este trabajo una iniciativa de implementación de una novedosa propuesta cuyo objetivo es el de incorporar los conceptos de metodologías Agile dentro del ámbito académico universitario. De esta manera pretendemos contribuir al proceso de alineación del proceso de enseñanza-aprendizaje con las necesidades actuales de negocios y tendencias tecnológicas de desarrollo de software. Este trabajo detalla la implementación de nuestra iniciativa en la asignatura de Trabajo de Campo de la carrera Ingeniería en Sistemas Informáticos de la Facultad de Tecnología Informática de la Universidad Abierta Interamericana sede regional Rosario. En nuestra propuesta realizamos la aplicación de metodologías Agile con Scrum como marco de trabajo dentro del proceso de enseñanza-aprendizaje de desarrollo de software, con la obtención de un módulo de software como resultado. En este trabajo evaluamos una serie de métricas dentro de un ámbito real para valorar fehacientemente la aplicación de nuestra propuesta y brindamos conclusiones y líneas de trabajo futuras para su efectiva implementación.

Este trabajo está organizado de la siguiente manera. La Sección 2 presenta la metodología Agile como concepto y como marco de trabajo para nuestra propuesta. La Sección 3 introduce nuestra propuesta de implementación de metodologías Agile en el ámbito académico. La Sección 4 detalla la implementación de la propuesta. La Sección 5 muestra novedades destacadas encontradas en la implementación de la propuesta. La Sección 6 describe conclusiones preliminares en la aplicación de la propuesta. Finalmente, la Sec-

---

<sup>2</sup> Una de las empresas es parte de la lista de Fortune 500 (<http://fortune.com/global500/>, último acceso mayo 2019). Omitiremos los nombres y los detalles de las mismas por motivos de confidencialidad comercial.

ción 7 brinda conclusiones generales sobre el trabajo y la experiencia realizada.

## 2 Metodología Agile. Un concepto y un marco de trabajo

Desde hace muchos años, existe un creciente interés por mejorar los modelos de desarrollo de software [12], [13]. En este camino de búsqueda continua, Kent Beck se reunió en el año 2001 con un conjunto de críticos y estudiosos del software para debatir sobre las técnicas y procesos de desarrollo aplicados hasta el momento, intentando reducir los problemas asociados a la rigidez en la aplicación de los mismos. De este trabajo conjunto se obtuvo un conjunto de 12 principios sobre los cuales se creó un documento denominado manifiesto Agile [14]. Este documento representa un marco teórico de trabajo que permite acortar los tiempos de desarrollo, eliminar la incertidumbre, mejorar la eficiencia en la producción y la calidad de los productos finales, tener capacidad de respuesta al cambio y brindar la mayor satisfacción posible al cliente a través de la entrega temprana y la retroalimentación continua durante la construcción del producto. Se enfoca a proyectos inciertos donde, a priori, no se dispone de toda la información necesaria como para planificar un desarrollo completo. A diferencia de otras metodologías, Agile no marca las pautas de cómo deben hacerse las cosas, sino que sienta las bases de una forma de trabajo sustentada sobre 4 postulados fundamentales. Los postulados son los siguientes: a) individuos e interacciones sobre procesos y herramientas; b) software funcionando sobre documentación extensiva; c) colaboración con el cliente sobre negociación de contratos; d) responder al cambio sobre seguir el plan.

Con respecto a los 12 principios sobre los que se desarrolló el manifiesto, consideramos importante mencionarlos aquí, dado que algunos de ellos servirán como base para el armado de nuestra propuesta. Estos principios son entonces detallados a continuación.

1. *Satisfacer al cliente* a través de la entrega temprana y continua de software de valor. Esta premisa es prioritaria dentro de la metodología Agile.

2. *Son bienvenidos los requisitos cambiantes*, incluso si llegan tarde al desarrollo. Los procesos Agile se doblan al cambio como ventaja competitiva para el cliente.

3. *Iteraciones constantes*. Entregar con frecuencia software que funcione, en periodos de un par de semanas hasta un par de meses.

4. *Trabajo colaborativo*. Las personas del negocio y los desarrolladores deben trabajar juntos de forma cotidiana a través del proyecto.

5. *Motivación del equipo*. Construcción de proyectos en torno a individuos motivados, dándoles la oportunidad y el respaldo que necesitan y procurándoles confianza para que realicen la tarea.

6. *Contacto directo con los clientes*. La forma más eficiente y efectiva de comunicar información de ida y vuelta dentro de un equipo de desarrollo es mediante la conversación cara a cara.

7. *Medida de progreso*. En este sentido, el software que funciona es la principal medida del progreso.

8. *Desarrollo sostenido*. Los procesos Agile promueven el desarrollo sostenido. Los patrocinadores, desarrolladores y usuarios deben mantener un ritmo constante de forma indefinida.

9. *Búsqueda de la excelencia*. La atención continua a la excelencia técnica enaltece el espíritu Agile.

10. *La simplicidad*, como arte de maximizar la cantidad de trabajo que se hace, es esencial.

11. *Autorregulación*. Las mejores arquitecturas, requisitos y diseños emergen de equipos que se autoorganizan.

12. *Revisión permanente*. En intervalos regulares, el equipo reflexiona sobre la forma de ser más efectivo y ajusta su conducta en consecuencia.

Así como la propia cultura Agile se renueva constantemente con el objetivo de mejorar las técnicas, existen diferentes metodologías Agile para el desarrollo de proyectos sin importar la complejidad o tamaño del mismo. Estas metodologías se pueden desarrollar en diversos ámbitos, no obstante lo cual cada una cuenta con diferencias en sus procesos. En oportunidad de acercar la metodología al ámbito académico, el foco está centrado en Scrum, enfoque de trabajo que aplica de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Se desarrolla por diferentes roles o figuras dentro del proceso, las cuales tienen distintas funciones [15]. Scrum es un marco de trabajo en el que se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados rápidos. En estos entornos los requisitos pueden cambiar o estar poco definidos y son de fundamental importancia las cualidades de innovación, competitividad, flexibilidad y productividad. Es por esto que dentro del marco de trabajo Scrum se establecen elementos de trabajo tales como los roles, las ceremonias y los artefactos necesarios. Es posible entonces, por medio de estos elementos, lograr una correcta utilización y seguimiento de la metodología. A continuación se realiza una enumeración y breve descripción de los mismos

*Roles:*

- *Product Owner* (Cliente). Representante de todas las personas interesadas para conseguir una buena definición de los objetivos del proyecto.
- *Scrum Master* (Facilitador). Elimina impedimentos. Vela por la implementación y cumplimiento de la metodología.
- *Team* (Equipo). Conjunto de personas con perfil técnico que de manera conjunta desarrollan el producto.

*Ceremonias:*

- *Sprint Planning* (Planificación de la iteración). Ceremonia donde se planifican las tareas a realizar en el Sprint en curso.
- *Sprint* (Ejecución de la iteración). Intervalo de tiempo corto en donde se desarrolla el incremento de un producto.
- *Scrum Daily Meeting* (Reunión diaria de sincronización del equipo). Reunión de breve duración, en el que se sincronizan las actividades que están ocurriendo en el sprint, y la planificación de las siguientes actividades.
- *Sprint Review* (Demostración de los requisitos completados). Sesiones de demostración en los que se busca validar lo que se desarrolló durante el Sprint.
- *Sprint Retrospective* (Retrospectiva) Última ceremonia de un Sprint. Oportunidad para el equipo de inspeccionarse a sí mismo, y crear un plan de mejora para el siguiente Sprint.

*Artefactos:*

- *Product Backlog Item o PBI* (Requisito). Elemento de validación tal como especificaciones, requisitos o historias de usuario.
- *Product Backlog* (Lista de requisitos priorizados). Lista priorizada de PBI, que representan en conjunto la visión y expectativas del cliente respecto a objetivos y entregas del producto o proyecto. La lista parcial correspondiente a un Sprint se denomina *Sprint Backlog*.
- *Burndown Chart* (Gráfico de trabajo). Herramienta gráfica para mostrar avance, desvíos y retrasos en cada Sprint de las historias comprometidas.

Los indicadores e instrumentos que fueron utilizados para el presente documento son los siguientes.

- *Asistencia a clases*. Se realizará un seguimiento de asistencia en clase de los alumnos a fines de evaluar la concurrencia clase a clase de tal modo de tener un indicador indirecto acerca de la adopción de la metodología por parte de los alumnos.

- *Retroalimentación por clase.* Evaluaremos la retroalimentación de los alumnos aportada desde dos aspectos: a) claridad en la explicación de la temática dada en la clase; b) grado de utilidad del contenido desarrollado en clase.
- *Temario de clase.* Esta es la planificación correspondiente a los contenidos dictados en clase sobre la metodología, organizados por fecha.
- *Burndown chart.* Este artefacto sirve como herramienta gráfica para ver el avance y estado del Sprint en curso. Permite identificar desvíos en la planificación del mismo.

### 3 Presentación de la propuesta

En este trabajo aplicamos de manera práctica una novedosa propuesta de implementación de metodologías Agile en el proceso de enseñanza-aprendizaje de una asignatura dentro de la carrera de Ingeniería en Sistemas Informáticos de la Universidad Abierta Interamericana. La propuesta consiste en aplicar los conceptos de metodologías Agile, detallados en la Sección 2, en el desarrollo de los contenidos de las asignaturas de Trabajo de Campo y Trabajo de Diploma, pertenecientes al tercer año de la carrera. En otro trabajo de los mismos autores [15], se realizó una encuesta de relevamiento sobre el conocimiento general de metodologías Agile con aplicación de Scrum dirigida a docentes de la carrera de Ingeniería en Sistemas Informáticos y a alumnos de tercer año de la mencionada carrera. Los resultados arrojados permitieron avanzar con nuestra propuesta, ya que encontramos que tanto la población docente como los alumnos no poseen conocimiento teórico o práctico de la metodología en cuestión. Por ende, la aplicación de la propuesta permitirá adquirir conceptos básicos de metodologías ágiles acortando así la brecha entre los conocimientos transmitidos por la universidad por sobre las expectativas del mercado actual. Cabe aclarar que la propuesta mencionada en este paper podría ser aplicada de manera extensiva en otras asignaturas donde debe alcanzarse un resultado luego de realizar un proceso colaborativo de trabajo como por ejemplo las asignaturas de quinto año (Seminario de Aplicación Profesional y Trabajo final de Ingeniería) pero previamente se debería capacitar al plantel docente en metodologías ágiles dado que manifestaron en la encuesta realizada no tener conocimientos en los fundamentos y la aplicación de dicha metodología. Se recomienda en particular comenzar con las asignaturas que tienen un objetivo tangible, tal como un código de software, para luego evaluar la factibilidad de su aplicación en otras asignaturas.

Para esta propuesta, se pretende llevar adelante un proceso de desarrollo cuyo resultado previsto es un módulo de software destinado a seguridad, gestionando roles, permisos y usuarios. En este proceso, las clases se constituirán

en espacios de encuentro entre los actores involucrados: docente, alumnos y dos de los autores de este trabajo. Cada uno de ellos jugará un rol particular. Los alumnos tendrán el rol de Team, mientras que el docente será el Product Owner. Por su parte, los autores oficiarán como facilitadores o Scrum Master. Los facilitadores son alumnos que se desempeñan como Scrum Masters dentro del sector privado.

Para dar un marco de trabajo, el cursado de la asignatura está estructurado en Sprints. Con un docente frente a la asignatura, 11 alumnos y 2 facilitadores se compone el esquema de trabajo áulico, junto con grupos de carácter individual. Dado que la asignatura tiene una duración de 16 clases semanales, establecimos que la duración de un Sprint es de 4 semanas, definiendo como el primer Sprint el denominado Sprint 0, en el cual nos encargamos de preparar el ambiente y definir los conceptos básicos. Desde el punto de vista de entrega o ejecución el Sprint 0 no debe ser tomado como sprint de desarrollo, sino que es una etapa de lanzamiento o configuración de la solución. Estos conceptos básicos se refieren a definir la herramienta de gestión a usar, los Sprints y la arquitectura a utilizar. En este Sprint no se provee un entregable en concreto para el Product Owner. De ahí en más se comienza a trabajar con los siguientes Sprints, numerados como 1, 2, 3 y 4. Al comienzo de cada Sprint, los equipos deben acordar junto al Product Owner la lista de requerimientos comprometidos a entregar al concluir el mismo.

Las clases tienen una extensión horaria de 4 horas por jornada semanal. Cada clase individual es estructurada en dos partes. En la primera parte de cada clase, con una duración aproximada de 2 horas, los facilitadores explican conceptos teóricos del enfoque Agile y en la segunda parte dichos conceptos son llevados al campo práctico completando las 4 horas del dictado de la cátedra. Es en esta última parte donde se aplican los conocimientos adquiridos y se los utiliza como herramientas para la revisión del Sprint en curso. De esta manera es posible llevar a cabo las ceremonias de la metodología (Daily meeting, Review meeting y Retrospective meeting) de acuerdo al momento del proceso. Mediante el seguimiento en clase por parte de los facilitadores, se controla el avance de cada Team por medio del Burndown chart. Con esto es posible ver el estado de los puntos comprometidos conforme avanza el Sprint, dejando visibles posibles demoras en la entrega. Como entregable los equipos deben presentar en formato impreso las historias de usuarios del Sprint finalizado junto con el avance de los puntos de las historias de usuarios. Este trabajo permite además hacer frente a posibles cambios en los requerimientos o evaluar una eventual renegociación con el Product Owner sobre la entrega de un Sprint.

El Team, junto con el Scrum Master, debe llevar a cabo la Daily Meeting en cada clase. Cabe aclarar que como las clases son semanales, dichas reuniones engloban lo trabajado durante la última semana, en lugar de comentar el avance diario como se hace usualmente en un contexto real. En la Daily Meeting cada integrante deberá responder a tres interrogantes. Las preguntas a contestar son: a) ¿En qué se estuvo trabajando dentro del Sprint en la última semana?; b) Con qué se va a continuar trabajando en la semana en curso?; c) Existe algún inconveniente o bloqueo que impida seguir con el trabajo planeado dentro del Sprint? Esta reunión no debe tomar más de 15 minutos, por lo se busca que cada integrante del equipo pueda resumir sus respuestas en 2 o 3 minutos. Como práctica descontracturante, implementamos esta reunión con sus integrantes de pie, pasando entre los mismos un elemento a modo de “posta” para establecer quién tiene la palabra. Se busca de esta forma que la reunión no se extienda demasiado en el tiempo. En esta reunión se completa el progreso de cada tarea, teniendo en cuenta el Burndown chart y evaluando si el trabajo del Team se mantiene acorde a la planificación inicial.

Con cada fin de un Sprint, se realiza en clase la ceremonia de Review Meeting. En esta ceremonia se realiza una demostración de las nuevas funcionalidades en la que se estuvo trabajando en dicho Sprint, con la participación de todos los actores. Posteriormente, en esta misma clase de cierre de Sprint, el Team junto con el Scrum Master realizan la ceremonia de Retrospective Meeting, cuyo fin es revisar el desempeño en conjunto que tuvo el equipo durante el último Sprint bajo tres puntos: a) ¿Qué tareas se realizaron bien?; b) ¿Qué se debe mejorar?; c) ¿Cuáles puntos de acción se deben cursar para mejorarlos? Dicha ceremonia no debe durar más de una hora u hora y media. Aquí se persigue el objetivo de mejorar de manera continua la productividad y la calidad del producto que está desarrollando, la motivación del Team, verificar cómo se están relacionando entre ellos, como fue la última iteración y cómo está yendo el proyecto. El Team analiza cómo ha sido su manera de trabajar durante la iteración, por qué está consiguiendo o no los objetivos a que se comprometieron al inicio de la iteración y por qué el incremento de producto que acaba de demostrar al Product Owner era o no lo que se esperaba. La evaluación se lleva a cabo de manera presencial ante el Product Owner y los facilitadores. Se evalúa de manera individual a cada alumno, verificando el avance en la herramienta en base a las historias de usuarios y la estimación de dicha historia de usuario realizada. Se pide exponer en la herramienta la escritura de la historia de usuario, criterios de aceptación y que se encuentre en el correcto estado dentro del Sprint. En paralelo se solicita al alumno que realice una breve demostración de la funcionalidad solicitada en la historia de usuario correspondiente.



A lo largo del proceso completo aplicaremos concretamente 5 de los 12 principios del manifiesto Agile, como se muestra a continuación.

*Principio 1: Entrega temprana.* El requisito de esta propuesta es completar el módulo de seguridad aplicando Scrum. Es por esto que en cada Sprint los equipos deben mostrar avances significativos ante el Product Owner, para que el mismo vea el progreso de los requerimientos de forma temprana. Por lo tanto, en cada clase cada grupo debe mostrar el avance en los requerimientos que se comprometieron en el respectivo Sprint para de esta forma tener una visibilidad del producto que se va a entregar, y evaluar si el trabajo se desarrolla según lo planificado o si hay algún desvío en el mismo.

*Principio 2: Requisitos cambiantes.* Si en algún momento el Product Owner decide agregar o quitar requerimientos, el equipo debe estar abierto al cambio, siempre y cuando el mismo no conlleve un enfoque totalmente distinto del producto final. Se propone entonces al docente que en un momento determinado, mientras se está transitando algún Sprint, genere a propósito algún cambio en un requerimiento. Con esta acción, se busca abordar lo que sucede en el mundo real cuando un Product Owner cambia las prioridades o los requerimientos. El equipo de trabajo debe entonces aceptar el cambio y planificarlo de manera que sea lo menos perjudicial posible y que no comprometa la entrega del producto al final del Sprint.

*Principio 5: Motivación del equipo.* Tener un equipo motivado ayuda a generar confianza entre los mismos para que las tareas se realicen de forma más eficiente. Darle autonomía, empoderar al equipo en el proceso de desarrollo y brindarle una devolución de desempeño constante, son acciones para tener motivado al equipo. La autonomía se genera permitiendo que el equipo trabaje de manera propia acompañando el desarrollo de software. Para esto se fortalece la idea de empoderamiento en el proceso de desarrollo, dándole potestad al equipo de crear su propia propuesta de trabajo para ir avanzando en el desarrollo cumplimentando los tiempos. En todo momento, la devolución del desempeño sirve como retroalimentación y le sirve al equipo para saber si esas decisiones ayudan a la evolución del proyecto o por el contrario hay que tomar acciones acordes al problema. Esta devolución se realiza clase a clase mediante instancias formales e informales. De manera formal a través de las ceremonias de Sprint Review o demostración de requisitos, donde se evalúa el avance del sprint junto con el docente y los facilitadores cuando finaliza un sprint. Por otro lado, de manera informal la devolución se realiza en la Scrum Daily Meeting en cada encuentro y Sprint Planning de manera verbal acompañando el avance de cada proyecto evacuando dudas y removiendo impedimentos si los hubiera.

*Principio 6: Contacto directo con el Cliente.* Es importante que el Team, ante cualquier consulta, lo hable inmediatamente con el Product Owner para generar de esta forma comunicación de ida y vuelta, y desarrollar un vínculo de confianza entre el Team y el Product Owner. Los Scrum Masters deben fomentar una comunicación bidireccional para que esté alineado tanto el Team como lo que se pretende del lado del Product Owner representado por el docente.

*Principio 12: Revisión permanente.* Es importante la revisión continua del sprint. Para llevar adelante dicha práctica, se aplica la ceremonia de Retrospective Meeting, que nos permite revisar qué fue lo que se hizo mal, para buscar cambiarlo o hacer énfasis en aquello que funcionó bien. La revisión del Sprint permite tener una visión clara de la evolución de desarrollo del equipo en términos de calidad. Esta ceremonia se debe llevar a cabo independientemente de si las cosas funcionaron bien o mal, ya que es una forma de realizar el balance del Sprint y poder aprender de los errores y aciertos del Team.

Finalmente, una vez transcurridos los Sprints con la estructura antedicha, y aplicando los principios mencionados, se espera obtener como resultado final un software terminado acorde a las necesidades del Product Owner. Con esta implementación, buscamos que desde el ámbito académico se logre alcanzar el valor agregado que promueve la metodología Agile. Este valor está fundado principalmente en lograr la satisfacción del Cliente y superar incluso las expectativas que él mismo tenía establecidas en un principio. Otra característica clave para el éxito de esta implementación es la posibilidad de adaptación rápida y constante al cambio. Por último, es importante señalar que el Cliente, al estar en contacto con el equipo en el trabajo diario, se siente parte de la solución.

#### 4 Implementación de la propuesta

Para la implementación de esta propuesta en un caso real tomamos la asignatura de Trabajo de Campo, en la cual nos encontramos trabajando hasta el momento de envío de este trabajo<sup>3</sup>. Dado que la asignatura corresponde al primer cuatrimestre del año 2019, los resultados obtenidos son parciales. No obstante esto, presentamos aquí la planificación realizada, así como los resultados obtenidos hasta el momento.

Clas	Fecha	Tema
1	2 de Abril	Clase no dictada por Feriado Nacional <sup>4</sup>

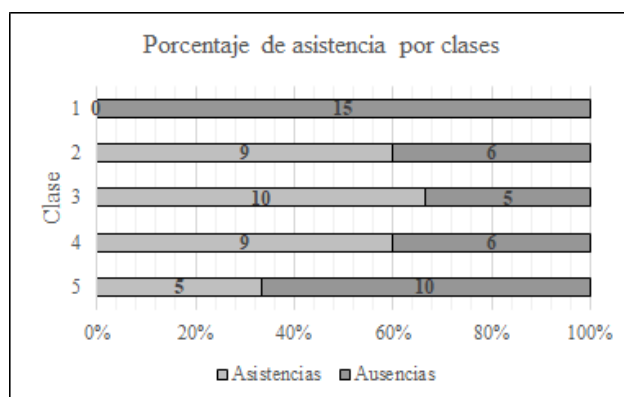
<sup>3</sup> Nos referimos a principios de mayo de 2019.

<sup>4</sup> Feriados nacionales: <https://www.argentina.gob.ar/interior/feriados>. Último acceso mayo 2019

2	9 de Abril	Introducción a la asignatura y Encuesta para Alumnos sobre Agile
3	16 de Abril	Conceptos de Scrum e introducción a la herramienta Taiga
4	23 de Abril	Métodos de estimaciones / Creación de historias de usuario detalladas
5	30 de Abril	Ceremonias Stand up meeting / Estimación historias de usuario / Sprint planning y Poker Planning <sup>5</sup>
6	7 de Mayo	Ceremonia de Stand up Daily meeting / revisión avance del proyecto
7	14 de Mayo	Ceremonia de Stand up Daily meeting / revisión avance del proyecto
8	21 de Mayo	Ceremonias de Stand up Daily meeting / Review meeting junto al PO/ Retrospective meeting
9	28 de Mayo	Ceremonias de Stand up Daily meeting / Sprint Planning
1	4 de Junio	Ceremonia de Stand up Daily meeting / revisión avance del proyecto
0		
1	11 de Junio	Ceremonias de Stand up Daily meeting / Review meeting junto al PO/ Retrospective meeting
1	18 de Junio	Ceremonia de Stand up Daily meeting / revisión avance del proyecto
2		
1	25 de Junio	Ceremonia de Stand up Daily meeting / revisión avance del proyecto
3		
1	2 de Julio	Ceremonias de Stand up Daily meeting / Review meeting junto al PO/ Retrospective meeting
4		
1	9 de Julio	Feriado
5		
1	16 de Julio	Ceremonia de Stand up daily meeting / revisión avance del proyecto
6		

**Tabla 1.** Temario de dictado de clases en la asignatura de Trabajo de Campo.

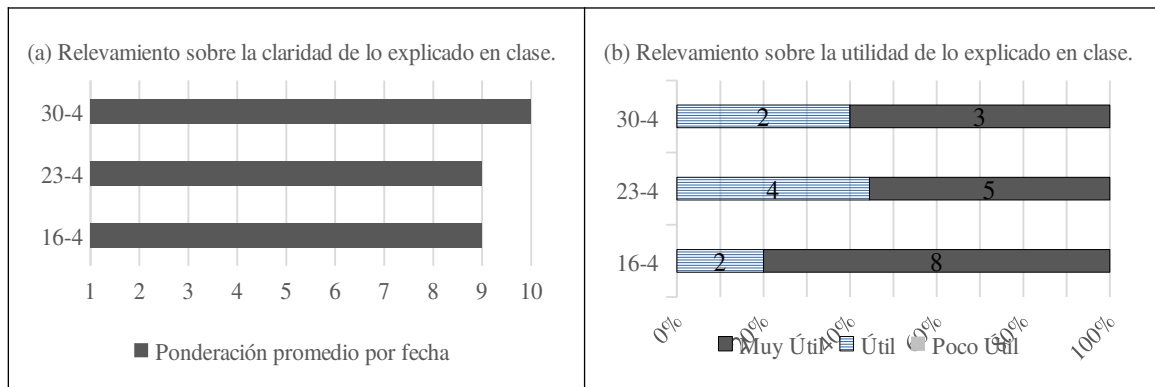
En la Tabla 1 se muestran los temas dictados al día de la fecha y planificados a ver en clase en este primer cuatrimestre del año lectivo 2019. Los conceptos vistos al 30 de Abril tienen un orden establecido para que los alumnos puedan dar comienzo a sus proyectos con los conocimientos mínimos de la metodología Agile en Scrum. A continuación detallaremos los resultados obtenidos de acuerdo a distintas métricas llevadas a cabo con el proceso general de la implementación de la propuesta



**Figura 1.** Asistencia de alumnos en las clases dictadas del calendario académico 2019

<sup>5</sup> Poker Planning: es una técnica para calcular una estimación basada en el consenso, en su mayoría utilizada para estimar el esfuerzo o el tamaño relativo de historias de usuario.

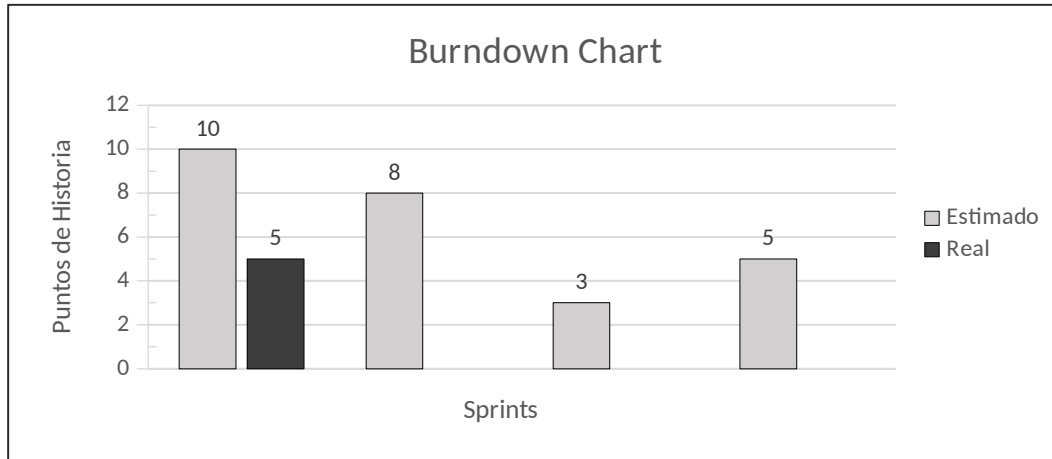
En la Figura 1 se muestra el resultado del relevamiento de la asistencia de los alumnos en las primeras 5 clases del ciclo lectivo en curso 2019. En la primera clase se registra una ausencia total dado que la fecha coincidió con un feriado nacional. Posteriormente, entre las clases 2 y 4 se observa un porcentaje igual o mayor al 60% de los alumnos. La última clase coincidió con una situación particular de medida de fuerza gremial a nivel nacional, lo cual redundó en una menor asistencia de los alumnos. Es importante tener en cuenta que el factor de la asistencia es importante para ser considerado en términos de una adecuada apropiación de la experiencia de enseñanza-aprendizaje por los actores involucrados.



**Figura 2.** (a) Relevamiento sobre la claridad de lo explicado en clase; (b) Relevamiento sobre la utilidad de lo explicado en clase

La Figura 2, por su parte, muestra un relevamiento de los resultados obtenidos a partir de las explicaciones conceptuales en clase. En la Figura 2 (a) solicitamos a los alumnos una ponderación en una escala de 1 a 10 acerca de la claridad en la transmisión de los conocimientos, donde 1 representa que no fue claro en absoluto y 10 representa un conocimiento transmitido en forma totalmente clara. Podemos observar entonces que los conocimientos transmitidos en clase fueron claros para los alumnos, siendo las puntuaciones promedio iguales o superiores a 9 en todas las clases. Por lo tanto concluimos que los alumnos comprendieron el conocimiento que se pretendió transmitir. En la Figura 2 (b) pedimos a los alumnos una evaluación cualitativa acerca de la utilidad de los conocimientos transmitidos, seleccionando entre las opciones Poco útil, Útil y Muy útil. Podemos observar que la totalidad de los casos evaluaban los conocimientos con las opciones Útil y Muy Útil, con mayoría de esta última opción en todos los casos. En conclusión, encontramos que los alumnos perciben los conocimientos transmitidos como muy útiles y de importancia dentro de cada encuentro áulico.

Otro punto relevante a tener en cuenta para evaluar los contenidos explicados y aplicados en clase es que realizan exámenes de autoevaluación a completar por los alumnos para medir el grado de comprensión de los contenidos dados.



**Figura 3.** Burndown Chart de un proyecto en curso

En la Figura 3 podemos observar el Burndown Chart en forma de gráfico de barras. Este gráfico muestra el grado de avance cuantificado en términos de puntos de historia para cada Sprint en curso dentro de un proyecto. Es necesario recordar que los datos recolectados hasta el momento solamente cubren parte del primer Sprint. Por lo tanto, agregamos al gráfico los puntos de historia estimados hasta el final del proyecto. Se puede evidenciar entonces el nivel de progreso de los puntos de historias según lo planificado en la duración del Sprint. De acuerdo al gráfico, puede verse que las tareas asignadas dentro del Sprint actual tienen un grado de avance en el desarrollo del 50%. Es esperable que al finalizar el primer Sprint este grado de avance se acerque lo más posible a la totalidad del 100%, y que este nivel se mantenga a lo largo del resto de los Sprints.

De acuerdo a estas evaluaciones preliminares, vemos que la aplicación de nuestra propuesta se desarrolla con un excelente grado de aceptación, y si bien la asistencia de los alumnos es un factor externo difícil de controlar dentro del espacio académico, los resultados obtenidos muestran un avance promisorio y una posibilidad importante de finalizar exitosamente con la entrega del producto y la satisfacción de todos los actores involucrados.

## 5 Novedades destacadas en la implementación de la propuesta

En esta Sección pretendemos dejar constancia de una serie de novedades destacadas en la implementación de la propuesta, de acuerdo a la experiencia de los autores como facilitadores respecto de los alumnos y del intercambio de impresiones con el docente a cargo del curso.

En primer lugar, se comenta que un obstáculo importante encontrado anteriormente a la hora de trabajar en este tipo de proyectos era la extensa documentación requerida en un formato específico. En cursos anteriores esta documentación era solicitada antes de comenzar a trabajar en la solución propiamente dicha. De acuerdo al docente, esto usualmente impedía visualizar de forma temprana el resultado del esfuerzo y disminuía la motivación de los alumnos. Utilizando la metodología Agile, buscamos que se reduzca la documentación a lo más esencial y dar lugar a que los alumnos como equipo puedan, de forma planificada, generar entregables como parte de una solución. De esta forma proveemos a una reducción de la frustración asociada a la excesiva documentación inicial.

Por otra parte, un elemento a destacar en nuestra propuesta es el rol del facilitador que aparece con la metodología Agile. El facilitador, que en este caso particular oficia además de capacitador específico en la metodología, ayuda a resolver problemas de forma temprana dando un seguimiento constante destrabando problemas o bloqueos si los hubiese.

Otro novedoso elemento que encontramos sumamente útil para acercar el ambiente académico al mundo real es la figura del Product Owner. Este nuevo rol, que encuentra estratégicamente su materialización en la persona del docente, es quien evalúa la solución y el proceso correspondiente.

Un concepto original introducido en nuestra propuesta es el de requerimiento cambiante. Con la aparición intencional pero sorpresiva de este elemento logramos reflejar el devenir de un proyecto de una forma más parecida a la realidad actual. Con respecto a la forma de trabajo llevada a cabo anteriormente, vemos que este concepto sería difícil de trabajar, ya que conllevaría volver atrás con documentación ya entregada, lo cual reduciría nuevamente la motivación y atrasaría los tiempos de entrega.

Finalmente, es de remarcar un factor cualitativo presente en todas las clases, cual es el de la dinámica implementada en la clase. Es claro que esta dinámica grupal es inherentemente colaborativa, lo cual aumenta la motivación y propicia el trabajo en conjunto para cada avance del proyecto.

## 6 Conclusiones preliminares sobre la implementación

La implementación de la propuesta arrojó varios ítems de análisis acerca de cómo fueron desarrollándose las clases. Con respecto a las 16 clases correspondientes al cronograma inicial de cursada pactadas por el calendario académico, vemos que 2 son afectadas a feriados nacionales y de las 14 restantes, solo se llegaron a dictar 4 hasta el momento de la presentación de este trabajo. Sobre el presentismo de los alumnos, podemos decir que en general tuvimos una concurrencia del 60% mínimo en cada clase, lo cual fue una asistencia baja, pero que no perjudicó la implementación efectiva de la metodología, ya que los alumnos faltantes incorporaron rápidamente el conocimiento de las clases perdidas bajo la intervención de los facilitadores y el docente.

El ambiente que se generó fue ameno en todas las clases. La participación y la constante devolución de todo el equipo de trabajo (Team, Scrum Masters y Docente) aportaron mucha riqueza en el contenido de cada clase. Cabe destacar que para llevar a cabo un proyecto en Agile aplicando Scrum, los facilitadores intentaron en todo momento que no existan jerarquías y dejaron en claro que ante un inconveniente la responsabilidad es asumida por todos como equipo. Estos conceptos afianzaron la dinámica de trabajo en equipo para los alumnos.

Cada uno de los indicadores evaluados clase a clase permitieron obtener conclusiones respecto de la implementación de metodologías Agile dentro de un entorno educativo para la materia de Trabajo de Campo. Se prevé continuar con la evaluación de estos indicadores durante el resto del cursado para llegar a una evaluación global al finalizar el mismo. Los resultados preliminares sobre los indicadores evidencian una participación e interacción exitosa de todos los actores involucrados, dada la gran aceptación y entendimiento de los conocimientos transmitidos.

Los facilitadores encontraron que los alumnos mantuvieron un nivel alto de motivación desde el comienzo del proyecto, pudiendo comprobar de manera temprana el grado de avance del proyecto, evitando la excesiva documentación y adquiriendo conceptos teóricos novedosos y concisos.

## 7 Conclusiones generales

Luego de haber iniciado la implementación de nuestra propuesta de incorporar metodologías Agile en el ámbito académico, dentro del proceso de enseñanza-aprendizaje en la asignatura de Trabajo de Campo de la carrera de Ingeniería de Sistemas Informáticos de la Universidad Abierta Interamericana sede regional Rosario, podemos concluir que los alumnos se ubican en un mejor posicionamiento frente al mundo laboral profesional actual, en donde mayormente se aplica este tipo de metodología de manera ampliada.

Al momento de la presentación, esta implementación se encuentra en curso, desarrollándose en las clases en la asignatura mencionada anteriormente. Este proceso sigue vigente dado que la asignatura tiene una duración cuatrimestral y el cursado tiene su cierre a fines de Julio. No obstante esto, dado el avance evidenciado hasta el momento, puede concluirse que dicha metodología constituye un valor agregado a la formación de los alumnos presentándose como una mejora en el proceso de enseñanza-aprendizaje de esta asignatura.

Los contenidos planificados hasta el momento dentro del cronograma de clases pudieron dictarse y ser comprendidos de manera satisfactoria, permitiéndoles a los alumnos comenzar el desarrollo de un módulo software de seguridad, aplicando metodologías Agile bajo una herramienta software de gestión de proyectos que permite revisar en todo momento el estado del mismo. Asimismo, se generó una dinámica de trabajo positiva, donde la transparencia entre los alumnos bajo el rol de Team y el docente bajo el rol de Product Owner ayudó a fomentar la confianza y el compromiso mutuo. Por su parte, la eliminación de documentación superflua, de acuerdo con el espíritu Agile, evitó demoras en las entregas parciales del producto al cliente.

Creemos firmemente que la aplicación de este proyecto sirve como puntapié inicial para otorgar una contribución a la Universidad, de tal forma que sume a su objetivo de moldear sólidos profesionales formados en las diferentes tecnologías y metodologías que son requeridas en forma efectiva por el mercado laboral actual.

Para el segundo cuatrimestre del presente año, se prevé la continuación de la implementación de nuestra propuesta con la asignatura Trabajo de Diploma. En esta segunda parte del año utilizaremos los resultados obtenidos para plantear otro trabajo de investigación que incorpore la implementación de metodologías Agile y estudie su adecuación al nuevo enfoque de educación centrada en el estudiante, de acuerdo a los nuevos estándares CONFEDI para carreras de ingeniería, con aplicación del importante concepto de Competencias.

## Referencias

- [1] J. Pajares, D. Poza, F. Villafañez, and A. López-Paredes, "Project Management Methodologies in the Fourth Technological Revolution," in *Advances in Management Engineering*, Springer, 2017, pp. 121–144.
- [2] M. E. Fayad, M. Laitinen, and R. P. Ward, "Software engineering in the small," *Commun. ACM*, vol. 43, no. 3, pp. 115–118, 2000.
- [3] J. A. Livermore, "Factors that Significantly Impact the Implementation of an Agile Software Development Methodology,," *JSW*, vol. 3, no. 4, pp. 31–36, 2008.
- [4] G. Kumar and P. K. Bhatia, "Impact of agile methodology on software development process," *Int. J. Comput. Technol. Electron. Eng.*, vol. 2, no. 4, pp. 46–50, 2012.



- [5] K. Kaur and A. Jajoo, "Applying agile methodologies in industry projects: Benefits and challenges," in *2015 International Conference on Computing Communication Control and Automation*, 2015, pp. 832–836.
- [6] J. Rosario, "La Tecnología de la Información y la Comunicación (TIC). Su uso como Herramienta para el Fortalecimiento y el Desarrollo de la Educación Virtual," *Dispon. en el Arch. del Obs. para la CiberSociedad en <http://www.cibersociedad.net/archivo/articulo.php>*, 2005.
- [7] C. Belloch, "Las Tecnologías de la Información y Comunicación en el aprendizaje," *Recuper. el*, vol. 25, 2012.
- [8] M. Area Moreira, "La innovación pedagógica con TIC y el desarrollo de las competencias informacionales y digitales," *Rev. Investig. en la Esc.* 64, 5-17., 2008.
- [9] J. M. Marco-Simó, M. J. Marco Galindo, and D. Riera Terrén, "Casi le dimos la vuelta a la enseñanza del desarrollo del software," in *JENUI 2015*, 2015, pp. 341–348.
- [10] A. Dagnino, "Increasing the effectiveness of teaching software engineering: A university and industry partnership," in *2014 IEEE 27th Conference on Software Engineering Education and Training (CSEE&T)*, 2014, pp. 49–54.
- [11] S. Moore, "Gartner Identifies Top 10 Data and Analytics Technology Trends for 2019," 2019.
- [12] W. W. Royce, "Managing the development of large software systems: concepts and techniques," in *Proceedings of the 9th international conference on Software Engineering*, 1987, pp. 328–338.
- [13] M. Brhel, H. Meth, A. Maedche, and K. Werder, "Exploring principles of user-centered agile software development: A literature review," *Inf. Softw. Technol.*, vol. 61, pp. 163–181, 2015.
- [14] K. Beck *et al.*, "Manifesto for Agile Software Development," *The Agile Alliance*, 2001. .
- [15] "Una novedosa propuesta de implementación de Metodologías Agile en el proceso de enseñanza-aprendizaje de una asignatura de Ingeniería en Sistemas Informáticos (Artículo en evaluación)," *Algunos de los autores son los mismos que los del Presente Trabajo. Los nombres de los autores no se proveen para permitir su correcta evaluación.*